

IN THE CLAIMS

1. (Previously Amended) An interactive voice response system comprising:
 - a compiler operative to compile documents retrieved by a fetcher into compiled document data in executable form;
 - a cache which stores the compiled document data prior to receipt of audio input from a given user requesting a text-based document; and
 - an execution thread that executes compiled document data retrieved from the cache by the fetcher.
2. (Previously Amended) A system according to claim 1 also comprising a storage device, which stores state information related to execution of said compiled document data.
3. (Previously Amended) A system according to claim 2 also comprising a backup VoiceXML Interpreter communicating with said storage device, the backup interpreter providing a response to a user in the event of a failure associated with a primary voice response system.
4. (Previously Amended) A system according to claim 3, wherein said storage device comprises a memory database external to said backup VoiceXML Interpreter.
5. (Previously Amended) An interactive voice response system comprising:
 - a fetcher operative to retrieve a compiled document;
 - a storage device which stores state information related to execution of said compiled document; and
 - a backup Voice XML interpreter in communication with said storage device utilizing the state information to execute the compiled document in the event of a failure.

6. (Cancelled)
7. (Previously Amended) A system as in claim 5, wherein said storage device is a memory database external to said backup VoiceXML interpreter.
8. (Cancelled)
9. (Cancelled)
10. (Cancelled)
11. (Previously Amended) In an interactive voice response system, a method comprising:
 - retrieving documents encoded according to VoiceXML;
 - compiling the retrieved documents into compiled document data in executable form;
 - caching the compiled document data for later retrieval and execution.
12. (Previously Amended) A method as in claim 11 further comprising:
 - storing state information related to execution of said compiled document data.
13. (Previously Amended) A method as in claim 12 further comprising:
 - providing a backup VoiceXML Interpreter that utilizes the stored state information to support continued service in the event of a failure.

- 4 -

14. (Previously Amended) In an interactive voice response system, a method comprising:
 - retrieving executable code derived from a text-based document;
 - and
 - storing state information related to execution of the compiled document; and
 - providing the state information to a backup VoiceXML interpreter to otherwise provide a response to a user in the event of a failure.
15. (Cancelled)
16. (Cancelled)
17. (Cancelled)
18. (Cancelled)
19. (Cancelled)
20. (Previously Added) An interactive voice response system configured as a server that provides requested audio information associated with text-based documents, the server comprising:
 - an execution thread that processes an incoming request and, based on an audio input from a given user, identifies a request for audio information associated with a text-based document; and
 - a fetcher that receives a signal from the execution thread to search a cache for executable code associated with the requested audio information, the fetcher retrieving corresponding executable code from the cache for execution by the execution thread to satisfy the request for audio information associated with the incoming request.

21. (Previously Added) A server as in claim 20 further comprising:
a compiler that converts the text-based document into executable speech code for storage in the cache prior to receipt of the incoming request.
22. (Previously Added) A server as in claim 20, wherein the fetcher initiates communication with a remote server to retrieve a text-based document associated with the requested information if corresponding executable code is not stored in the cache.
23. (Previously Added) A server as in claim 22 further comprising:
a compiler that converts retrieved text-based documents into executable code for storage in the cache.
24. (Previously Added) A server as in claim 20, wherein executable code stored in the cache is concurrently utilized by multiple execution threads to provide a response to multiple users.
25. (Previously Added) A server as in claim 20 further comprising:
a storage device to store state information related to the executable code executed by the execution thread to satisfy the request for audio information associated with the incoming request.
26. (Previously Added) A server as in claim 20, wherein the executable code retrieved from the cache is associated with a corresponding viewable text-based document available on the World Wide Web.
27. (Previously Added) A server as in claim 20, wherein the execution thread receives the incoming call from a switchboard.

28. (Previously Added) A server as in claim 20 configured as a VoiceXML interpreter.
29. (Previously Added) A server as in claim 20 coupled to a data base to store state information associated with executable code being executed by the execution thread, the state information accessible by a backup VoiceXML interpreter to provide service in the event of a failure.
30. (Previously Added) A method of providing requested audio information associated with text-based documents, the method comprising:
 - processing an incoming call based on an audio input from a given user;
 - identifying a request for audio information associated with a text-based document;
 - searching a cache for executable code associated with the requested audio information, the executable code generated in response to a previous request from another user for audio information associated with the text-based document; and
 - executing corresponding executable code from the cache to satisfy the request for audio information associated with the incoming call.